

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A method for controlling trap generation of ~~[[an]]~~ a SNMP (Simple Network Management Protocol) ~~which is operated between a manager and at least one agent, wherein comprising:~~

defining in a MIB (Managed Information Base) a TrapFlag field and a TrapPeer field ~~are defined for each management-object resource (each object) in describing an MIB of an SNMP and more than~~ and including at least two objects are correlated together to define a trap generation condition,

wherein a value of the TrapFlag field is set by the manager to indicate whether or not the agent should issue a Trap message for the corresponding object, and

wherein a value of the TrapPeer field is set by the agent to indicate whether or not the trap generation condition is satisfied.

2. (Canceled).

3. (Currently Amended) The method according to claim 1, wherein ~~[[a]]~~ the value of the TrapFlag field is set ~~[[as]]~~ in an 'ON' state or 'OFF' state by the manager.

4. (Currently Amended) The method according to claim 3, wherein the agent generates a trap for a corresponding object ~~in case that~~ when the TrapFlag field is in the 'ON' state, and while, the agent does not generate a trap ~~in case that~~ when the TrapFlag field is in the 'OFF' state.

5. (Canceled).

6. (Currently Amended) The method according to claim 5, wherein the TrapPeer field is set ~~[[as]]~~ in an 'ON' state or an 'OFF' state by the agent.

7. (Currently Amended) The method according to claim 6, wherein the agent sets the TrapPeer field ~~[[as]]~~ in the 'ON' state ~~in case that the~~ when a state of an object satisfies a trap generation condition.

8. (Canceled).

9. (Currently Amended) A method for controlling trap generation of an SNMP (Simple Network Management Protocol) ~~which is~~ operated between a manager and at least one agent, comprising ~~the steps of:~~

defining a TrapFlag field and a TrapPeer field in ~~[[an]]~~ a MIB (Managed Information Base) of an agent;

setting a TrapFlag field value according to ~~the~~ a message outputted from the manager;

setting a TrapPeer field value for each object by the agent according to ~~[[the]]~~ a Trap generation condition defined in the MIB; and

generating a trap for an object according to the values of the TrapFlag field and the TrapPeer field.

10. (Original) The method according to claim 9, wherein the TrapFlag field is a field for indicating whether a trap is to be generated for each object described in the MIB.

11. (Original) The method according to claim 9, wherein the TrapPeer field is a field for defining a trap generation condition for an object.

12. (Currently Amended) The method according to claim 9, wherein the agent sets the TrapPeer field ~~[[as]]~~ in the 'ON' state ~~in case that~~ when the state of an object satisfies a trap generation condition.

13. (Currently Amended) The method according to claim 9, wherein the trap generation condition is defined by correlating ~~more than~~ at least two objects.

14. (Currently Amended) The method according to claim 9, wherein generating the trap ~~generating step comprising the sub-steps of~~ comprises:

searching ~~[[a]]~~ the TrapFlag field of each object when it comes to a trap generation period;

checking ~~[[the]]~~ a state of ~~[[a]]~~ the TrapPeer field ~~in case that~~ when the TrapFlag is in an 'ON' state; and

generating a trap for a corresponding object ~~in case that~~ when the TrapPeer is in the 'ON' state.

15. (Currently Amended) The method according to claim 14, wherein ~~in case that~~ when the ~~TrapFlag~~ TrapFlag is in the 'OFF' state, a trap is not generated ~~in no case~~.

16. (Currently Amended) The method according to claim 14, wherein ~~in case that~~ when the TrapFlag is in the 'ON' state and the TrapPeer is in the 'OFF' state, a trap is not generated.

17. (Currently Amended) A method for controlling trap generation of an SNMP (Simple Network Management Protocol) ~~comprising the steps of~~:

defining a TrapFlag field and a TrapPeer field in ~~[[an]]~~ a MIB (Managed Information Base) of an agent; and

generating a trap for an object according to the values of the defined TrapFlag field and the TrapPeer field ~~as defined, of which,~~

~~the step of wherein~~ generating ~~[[a]] the trap comprising the sub-steps of~~
comprises:

searching the TrapFlag field of each object when it comes to a trap generation period; ~~[[c]]~~

~~[[hecking]]~~ checking a state of the TrapPeer field ~~in case that~~ when the TrapFlag field is in an ON state; and

generating a trap for a corresponding object ~~in case that~~ when the TrapPeer field is in ~~[[an]]~~ the ON state.

18. (Original) The method according to claim 17, wherein the TrapFlag field is a field for indicating whether a trap is to be generated for each object described in the MIB.

19. (Original) The method according to claim 17, wherein the TrapPeer field is a field for defining a trap generation condition for an object.

20. (Currently Amended) The method according to claim 17, further comprising ~~the steps of:~~
setting ~~[[a]]~~ the TrapFlag field value according to a message outputted from the manager; and

setting [[a]] the TrapPeer field value for each object in the agent according to ~~the~~ trap generation conditions defined in the MIB.

21. (Currently Amended) The method according to claim 20, wherein the manager outputs a message to set [[a]] the TrapFlag field of an object of which state is not changed [[as]] in an 'OFF' state.

22. (Currently Amended) The method according to claim 20, wherein the trap generation condition is defined by correlating ~~more than~~ at least two objects.

23. (Currently Amended) The method according to claim 20, wherein the agent sets the TrapPeer field [[as]] in the 'ON' state ~~in case that~~ when the state of an object satisfies a trap generation condition.

24. (New) A SNMP (Simple Network Management Protocol) management system, comprising:

a MIB (Managed Information Base) including definitions for a TrapFlag field and a TrapPeer field for each management-object resource to define a trap generation condition;

an agent configured to set a value of the TrapPeer field to indicate whether or not the trap operation condition is satisfied; and

a manager connected to the agent and configured to set a value of the TrapFlag field to indicate whether or not the agent should issue a Trap message for the corresponding object when the trap condition is satisfied.

25. (New) the system of claim 24, wherein the generation condition includes at least two objects correlated together and which must be met to satisfy the trap generation condition.

26. (New) The system according to claim 24, wherein the agent generates a trap for a corresponding object when the TrapFlag field is in the 'ON' state, and the agent does not generate a trap when the TrapFlag field is in the 'OFF' state.

27. (New) The system according to claim 24, wherein the agent sets the TrapPeer field is an 'ON' state when a state of an object satisfies the trap generation condition.